CTHULHU: The Design and Implementation of Duke Robotics Club's 2019 AUVSI Competition Entry

Samuel Rabinowitz, Eric Chang, David Miron, Estelle He, Neil Dhar,
Christopher Wolff, Trevor Fowler

***Abstract*— In 2019, the Duke Robotics Club returns for their third year in recent history to the AUVSI RoboSub Competition with a brand new, smaller, leaner, and more maneuverable robot: Cthulhu. Successor to the team's previous robot, Leviathan, Cthulhu is the brainchild of the dozens of Duke engineers that make up the student-run Duke Robotics Club. Work on Cthulhu, split between three different subsystem teams (subteams), began in fall of 2018. The new robot features a more compact design (2/3 the size of Leviathan), a redesigned electronics stack, and modularity for greater interaction with the environment. Additionally, the software team successfully transitioned from an unmaintainable legacy codebase to Docker and ROS, which are industry standards. The team is also an active member of the Durham, NC community, mentoring high school FTC and FRC teams.**

I. Introduction

The Duke Robotics Club is a student led project team from Duke University's Pratt School of Engineering. The team is made up of members from math, science, and engineering. The team competed in RoboSub in 2006 and 2007. After a hiatus from the competition, during which the team devoted time to various other projects in collaboration with Duke professors, the team returns to RoboSub for the third time since 2016.

The mechanical subteam was responsible for the frame design, electronics enclosures, and actuators. Cthulhu's waterproof mechanical design is the result of extensive research, simulation, modeling, and in-house CNC machining, leading to a robust capsule that has not leaked in more than 50 of hours of testing.

The electronics subteam was responsible for the power architecture, sensing systems, onboard computation hardware, and microprocessor firmware. The largest success this year was simplifying Cthulhu's capsule down to 2/3 of the previous size while also extending the capabilities and modularity to create a fully featured platform for informing and running the codebase.

The software subteam wrote software to control the robot, handling everything from sensor fusion to motion planning to computer vision. Additionally, the software team successfully transitioned from an unmaintainable legacy codebase to a more maintainable Docker and ROS based system, adopting industry standards.

II. Competition and Design Strategy

Heading into the 2018-2019 year, Duke Robotics Club had strong momentum from the previous vanguard of graduates, but current membership was young. The team's former Leviathan was due for a replacement, so a strategic balance was adopted of building knowledge and expertise in newer members while tackling a new robot.

The majority of robot design occurred in the fall semester. Learning from Leviathan, the team focused on reliability, robustness, and maneuverability over complexity and added functionality.

As such, and because tasks had not yet been announced, the team placed an emphasis on modularity, which allowed for expansion into more complex tasks later without stalling preliminary development. Not only does this modularity allow for use in future years, it also allows the bot to be a testing bed for new modules that will ultimately end up on new robots.

A. Mechanical Subteam Strategy

Physical design began with whiteboarding, moving then to computer-aided design (CAD) in SolidWorks, then 3D printing, and finishing with in-house machining, often with computer numerical control (CNC). Several key structures underwent rigorous finite element analysis (FEA) in SolidWorks. In some cases, such as ensuring adequate buoyancy and balance, a mix of numerical analyses and tests with the partially-assembled robot helped fine-tune the robot's properties. Front-loading this design work prior to manufacturing minimized time-consuming and costly mistakes.

Once the primary mechanical tasks were well underway, the team addressed other competition tasks, focusing on the ones that bore the greatest similarity to those from prior years. Specifically, the marker dropper and torpedo tasks were nearly identical to previous competitions, so the team drew upon institutional knowledge to adapt and improve existing solutions. Modularity allowed for parallel component development: as competition information was released, task-specific modules were rapidly designed, prototyped, and tested in water without being fully integrated.

B. Electronics Subteam Strategy

The electronics subteam also focused on the overall goals of reliability, robustness, and modularity, and thus had the following major design constraints:

- Improve on prior functionality while shrinking the electronics stack to less than 2/3 of its previous size.
- Make an easily accessible, modifiable, and modular stack. This was designed in response to a lesson learned from Leviathan, which opted for a minimally-structured stack for ease of wiring, but proved challenging to troubleshoot and expand upon, causing major concerns for reliability.
- Reduce reliance on other subteams to test.

To these ends, the process of creating the electronics stack was completely reimagined. Rather than haphazardly wiring a variety of different components together, and only then attempting to fit them in the stack, a strategy similar to that of the mechanical team was adopted. All proposed components went through a rigorous review process, including ensuring their compatibility with other components and hours of testing under load. Lastly, the electronics subteam elected to use a Pixhawk controller to allow for testing independent of other subteams.

C. Software Subteam Strategy

The software subteam went through a radical shift this year to align with the new goals of reliability and modularity. For the last few years, the codebase, controls algorithms, mathematical models, and sensor fusion were developed in-house. This resulted in old, undocumented and unmaintainable code, so the subteam elected to make the transition to using Docker and Robot Operating System (ROS). ROS promotes increased modularity, easy communication with the Pixhawk, and reliability given the extensive testing it has gone through as an industry standard.

III. Vehicle Design: Mechanical System

Fig. 1. A rendering of Cthulhu's core frame and elements.

A. Capsule

One of the mechanical subteam's top priorities was designing a capsule that was significantly smaller and easier to open than that of Leviathan, as well as maintaining Leviathan's precedent of robustness and waterproofing. Leviathan's design used a double-O-ring static radial seal held in place by two screws, and the capsule was attached to the robot while the electronics stack was removable for maintenance. Stack removal was an arduous process: first unplug all connectors (contributing to mechanical wear and causing confusion when replugging), next unscrew two screws, then tighten them into different holes to break the O-ring seal, and finally pry the stack out of the capsule.

Cthulhu completely reimagined Leviathan's design. Now, the stack was mounted on the robot, and the capsule was removed from around it. Removing the capsule was as easy as popping two latches, removing two thumb screws, and sliding the capsule off. Connectors remained plugged in. The stack could now be maintained and tested while still on the robot. SolidWorks FEA informed structural changes to support the cantilevered stack. Polycarbonate was used for increased strength with less material. The same O-ring seal between flanged metal endcaps was carried over from Leviathan for proven reliability. When needed, the stack can be removed for extra maintenance by removing only four screws.

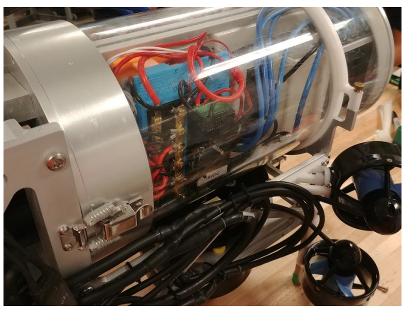

Fig. 2. Cthulhu's main capsule. Note the latch at left.

B. Frame

Cthulhu's frame was redesigned from the ground up to be simpler, lighter, and stronger than that of Leviathan. The key decision was to CNC mill and water-jet-cut two large aluminum sidings and two connecting mounts in-house, rather than screwing together off-the-shelf aluminum bars. Extensive SolidWorks FEA minimized the amount of aluminum present while providing necessary structural integrity. The resulting frame consists of only eight distinct parts and can be disassembled and reassembled in under 10 minutes.

Keeping in mind the design goal of modularity, additional space was added for new components in front of this year's smaller new capsule, below the capsule around the DVL, and on the sides of the frame between the smaller thrusters. HDPE mounts protect sensitive components from scratching, and the entire frame was anodized to prevent corrosion.

C. Electronics Stack (Mechanical Design)

A major point of success was the mechanical team's partnership with the electronics team to

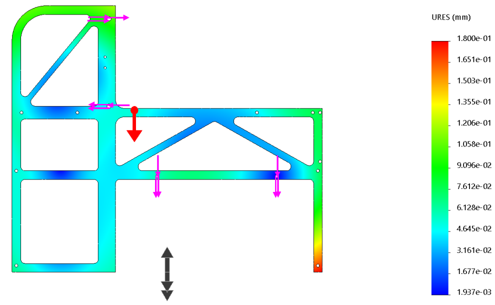

Fig. 3. One of two frame sides undergoing FEA.

3D print a customized network of mounts for the electronics stack, including specific channels for clean wiring. Component layout was optimized for quick accessibility to those requiring most troubleshooting. The result was a 1/3 smaller and more accessible, organized, and aesthetically pleasing stack. See Fig. 4.

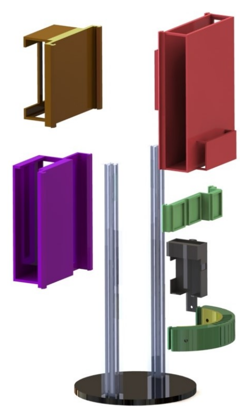

Fig. 4. Exploded view of electronics stack. A customized network of mounts holds all of the electronics, including specific channels for clean wiring. The stack attaches to the fixed end cap for capsule removal independent of the stack.

D. Actuators

1) Thrusters: Cthulhu uses smaller, lighter, and more economical Blue Robotics T200 thrusters, which replace older, heavier, and more costly SeaBotix thrusters. Eight vectored thrusters on custom-CNC'ed Delrin mounts allow Cthulhu to move with full six degrees of freedom. New thrusters and a smaller and lighter robot have yielded more maneuverability and reduced power consumption.

2) Marker Dropper: The marker dropper system used aboard Cthulhu allows for reloading of collected golf balls mid-mission. Modified from a pneumatic controlled to servo controlled design, the marker dropper works by sliding a box with a cylindrical hole from the input tube to the output, allowing a single marker to drop at a time. Many bots, including Leviathan, use magnets to release metal markers; Cthulhu's system can be actuated by a single servo and is lighter and smaller than two magnetic systems.

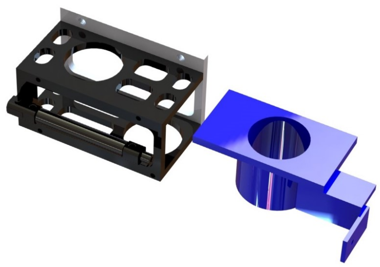

Fig. 5. Exploded view of marker dropper.

3) Torpedo Launchers: Cthulhu's torpedo launchers reduce the complexity and failure points of Leviathan's pneumatic system by opting for a spring-loaded, servo-controlled design, inspired by BBAUV 3.5's torpedo design [2]. The design features a double barrel setup that allows both spring-loaded torpedoes to be released via a single servo. The device mounts parallel to the frame to limit the intrusion of its long barrels to other robot elements. The team found that front-weighted torpedoes with one and four fins travelled the straightest and longest. See Fig. 6.

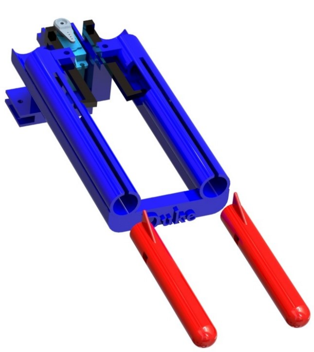

Fig. 6. A rendering of Cthulhu's torpedo launcher. The torpedoes are spring loaded, and each is fired separately via one servo.

IV. Vehicle Design: Electronics System

A. Electronics Stack

The electronics stack inside the capsule provides the critical infrastructure needed to route power and communications reliably between subsystems. This includes housing the robot's battery, computer, networking hardware, thruster controllers, and various sensors. To minimize the size and weight of the stack, Cthulhu incorporates just a single 16000mAh LiPo battery to power both the computer and thrusters. To provide a steady voltage source for the various different components while protecting sensitive electronics from unexpected power supply issues, the stack into two distinct power rails with their own fuses and relays. Each power rail can be independently operated, making it easy to minimize power consumption when only parts of the robot are being tested. A series of voltage regulators then adjust the battery supply voltage to meet the specification of individual components, including a 19V supply for the Intel NUC and a 48V supply for the PoE hardware. New this year, the NUC creates a network of its own, connecting to a land-side tether and the three cameras using an onboard Power over Ethernet switch.

B. Sensors

1) Cameras: A downward-facing camera tracks path markers and provides awareness of surroundings. Stereo front-facing cameras identify obstacles and generate 3D map (point-cloud) of the environment using Simultaneous Localization and Mapping.

2) Localization: A Doppler Velocity Logger allows determination of instantaneous velocity, which is integrated to acquire position with minimal error. An Inertial Measurement Unit inside of the Pixhawk provides corroborating acceleration data, and a Blue Robotics Bar30 pressure sensor determines depth.

3) Acoustics: In order to locate the acoustic pinger, an array of 4 miniature omnidirectional Teledyne TC4013 hydrophones was installed on Cthulhu. The data acquisition device, Saleae Logic 8, helps to sample the target signal simultaneously across 4 channels at a rate of 625k sample/second, which is more than 10 times of the maximum target frequency. This high resolution in sampled signal improves the accuracy in the phase comparison stage after the signal being passed through a 5^{th} order Chebyshev type II digital band-pass filter. Moving average and cross-correlation are used to determine the critical window for phase comparison and obtain the phase difference across 4 channels. With the phase difference data, Time Difference of Arrival approach is taken to locate several possible locations of the pinger in 3D space, and the resulting measurements help to eliminate the options and point to a stable result.

C. Thrusters and Controls

Finally, to perform low level control and power the thrusters, the stack also incorporates eight individual ESCs and a Pixhawk flight controller. Having each thruster controlled by a separate, smaller ESC gives flexibility in placement and thermal design and also keeps the design modular, so that thruster failure will not propagate.

For reliability, the new thrusters were tested extensively to ascertain their optimal operating voltage and maximum current draw. Underwater thruster connections were sealed not with the traditional potting epoxy but with heat shrink after thorough research revealed heat shrink's use in Navy submarines and other undersea vessels for increased reliability and handling. [6], [7]

V. Vehicle Design: Software System

A. Software Infrastructure

The new software infrastructure was designed with testing, reproducibility and efficient development in mind. One major pain point in previous years has been not only coordinating the development environment throughout the software team but also ensuring that changes and updates are communicated and deployed regularly. In order to resolve these issues, two very significant changes were made to the way in which the team's software is written and deployed.

First, the software stack has been rebuilt from the ground up using ROS, allowing usage of a robust, well-documented set of libraries. Switching over to ROS and following best practices means avoiding the transfer of most legacy code. While a setback in the short term, it is the best decision to build a stable software stack and foster collaboration in the long term.

Second, Docker has been employed to entirely containerize the code needed to run Cthulhu. Coordination of development has been eased by the portability of development environments that Docker provides, not to mention the Dockerfile as a source of documentation for future years.

B. Robot Localization

Robot localization uses the Extended Kalman Filter that is a part of the ROS package `robot_localization`. It is fed odometry data acquired from the Pixhawk and DVL.

C. Computer Vision

To detect and localize the various objects on the course, a convolutional neural network with residual connections, similar to ResNet [9] but much smaller, is used. The model is a mapping from the preprocessed camera image to a vector with indicator variables and bounding box coordinates for each object. Training data currently comes from the training pool, with plans to use images from the competition site for the final model. After initial training via adaptive momentum estimation [3], the model outputs are calibrated to represent confidence scores for each object.

Several techniques ensure stable training. A histogram of oriented gradients [1] that represent the image improves sample efficiency by drastically lowering dimensionality. Data augmentation techniques such as cropping, scaling, and additive noise help counteract the lack of sample availability. For Cthulhu's implementation, RectLabel [4] is used to create the dataset and PyTorch [5] to train and run the model.

VI. Experimental Results

A. Testing Process

Vigorous testing of the mechanical systems was an integral part of development, and was used throughout the construction process. The initial designs were guided by bounded SolidWorks simulations, placing theoretical bounds on the frame thicknesses needed to maintain integrity. SolidWorks FEA was also executed on frame and thruster-mount parts. Mechanical components were then stress tested early and often, with the capsule having completed 24-hour trials at over 15 feet underwater. Torpedo type and launcher stability were also tested extensively. The electrical components were also tested often, but full-system tests were intentionally kept to a minimum because of the disastrous damage a single leak could cause the electronics. Partial software testing was performed often in the lab, while acquiring data from the DVL, testing motor control, or testing motion planning in simulation. Full software tests were performed at the pool, identifying problems specific to water environments.

B. Results

At the time of this writing, Cthulhu is a bare-bones vehicle capable of full freedom of motion at roughly 2 knots translational speed. The robot has three functional cameras for identifying obstacles below and in front of it, three positioning sensors including a DVL for navigation, and a single pressure hull. The robot is capable of basic motion planning and acoustic localization, but currently has no fully functional external actuators for non-navigational tasks. A torpedo system is in place, but completion of that task is not implemented in the software. By the time of the competition, it is hoped that the robot will have gained obstacle interpretation through vision. It has enough battery capacity to run for around 30 minutes with medium duty-cycle thruster usage (30-50%).

C. Lessons Learned

While the team refined its process based on past experience, there was no dearth of lessons to be learned from such a complex undertaking. Here are some highlights:

- **Prototype, integrate, and test, early and often.** For example, the team was thrown off by having to balance the robot in the water in the first pool tests at the end of the semester. Early testing would have solved that.
- **Communicate and delegate work.** Delays were frequently not engineering-related, but rather due to miscommunication. Real progress was made when the team worked as a cohesive unit.
- **Promote knowledge transfer.** Having just a few people understand certain subsystems has been a failure point in past years. While strides were made to alleviate it, there is room for improvement. Furthermore, engaging more members by empowering them to make an impact benefits the team overall.

ACKNOWLEDGMENTS

Duke Robotics Club is housed within the Pratt School of Engineering. The team gratefully acknowledges the Duke faculty and administrative staff who have helped and continue to help make the club a success. Special thanks to Professors Michael Zavlanos and Kris Hauser, Pratt's Undergraduate Program Coordinator Jennifer Ganley, and the Engineering Alumni Council, especially the club's liaison, Mike Wesley. Lastly, the team owes its continued success to its long-time sponsors: The Lord Foundation, Duke Student Government Student Organization Funding Committee, General Motors, and SolidWorks.

REFERENCES

[1] Dalal, Navneet, and Bill Triggs. ”Histograms of Oriented Gradients for Human Detection.” https://lear.inrialpes.fr/people/triggs/pubs/Dalal-cvpr05.pdf.

[2] E.W. Goh, *et al*, ”Design and Implementation of BBAUV 3.5.” *Robosub 2018.* https://www.robonation.org/sites/default/files/NatlUSingapore Bumblebee_TDR_RS18.compressed.pdf.

[3] Kingma, Diederik P., and Jimmy L. Ba. ”ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION.” https://arxiv.org/pdf/1412.6980.pdf.

[4] ”Labeling Images for Bounding Box Object Detection and Segmentation.” RectLabel. Accessed July 09, 2019. https://rectlabel.com/.

[5] Pytorch. ”Pytorch/pytorch.” GitHub. July 09, 2019. Accessed July 09, 2019. https://github.com/pytorch/pytorch.

[6] United States. Department of the Navy. Naval Sea Systems Command. *MIL-STD-2003-5A*. pp. 13, 50. September 3, 2009. Accessed July 2, 2019. http://everyspec.com/MIL-STD/MIL-STD-2000-2999/MIL-STD-2003_5A_25950/.

[7] United States. Department of the Navy. Naval Sea Systems Command. *Updates to MIL-STD-2003*. By Christopher Nemarich. March 9, 2017. Accessed July 2, 2019. https://nsrp.org/wp-content/uploads/2017/03/04-Updates-to-MIL-STD-2003-NSRP-9-Mar-17-Final.pdf.

[8] ”LEVIATHAN: The Design and Implementation of Duke Robotics Clubs 2017 AUVSI Competition Entry.” 2017. Accessed July 2, 2019. https://robonation.org/sites/default/files/RS17_Duke_Paper.pdf.

[9] Zhang, Ren, Sun, and Jian. ”Deep Residual Learning for Image Recognition.” ArXiv.org. December 10, 2015. Accessed July 09, 2019. https://arxiv.org/abs/1512.03385.

Appendix A: Expectations

Subjective Measures			
	Maximum Points	Expected Points	Points Scored
Utility of team website	50	45	
Technical Merit (from journal paper)	150	135	
Written Style (from journal paper)	50	45	
Capability for Autonomous Behavior (static judging)	100	85	
Creativity in System Design (static judging)	100	90	
Team Uniform (static judging)	10	10	
Team Video	50	45	
Pre-Qualifying Video	100	0	
Discretionary points (static judging)	40	35	
Total	650	490	
Performance Measures			
	Maximum Points		
Weight	See Table 1 / Vehicle	84	
Marker/Torpedo over weight or size by <10%	minus 500 / marker	0	
Gate: Pass through	100	100	
Gate: Maintain fixed heading	150	150	
Gate: Coin Flip	300	0	
Gate: Pass through 60% section	200	0	
Gate: Pass through 40% section	400	400	
Gate: Style	+100 (8x max)	0	
Collect Pickup: Crucifix, Garlic	400 / object	0	
Follow the “Path” (2 total)	100 / segment	100	
Slay Vampires: Any, Called	300, 600	300	
Drop Garlic: Open, Closed	700, 1000 / marker (2 + pickup)	0	
Drop Garlic: Move Arm	400	0	
Stake through Heart: Open Oval, Cover Oval, Sm Heart	800, 1000, 1200 / torpedo (max 2)	0	
Stake through Heart: Move lever	400	0	
Stake through Heart: Bonus - Cover Oval, Sm Heart	500	0	
Expose to Sunlight: Surface in Area	1000	0	
Expose to Sunlight: Surface with object	400 / object	0	
Expose to Sunlight: Open coffin	400	0	
Expose to Sunlight: Drop Pickup	200 / object (Crucifix only)	0	
Random Pinger first task	500	0	
Random Pinger second task	1500	0	
Inter-vehicle Communication	1000	0	
Finish the mission with T minutes (whole + factional)	Tx100	0	

Appendix B: Component Specifications

Component	Vendor	Model /Type	Specs	Cost (if new)
Buoyancy Control	N/A			
Frame	N/A	8021 Aluminum, custom		~$500
Waterproof Housing	N/A	Polycarbonate, custom		$1100
Waterproof Connectors	Subconn and Seacon wet-mate connectors			
Thrusters	Blue Robotics	T200	T200-THRUSTER-R1-RP	8 x $169
Motor Control	Blue Robotics	Basic ESC		8 x $25
High Level Control	Blue Robotics	Pixhawk	PIXHAWK-R1-RP	$120
Actuators	Hitec	D646WP waterproof servo		2 x $55
Propellers	included with T200 thrusters			
Battery	Turnigy	HC 4S 12C 16000mAh Lipo		$108.80
Converter	Kohree	DC/DC 36V/12V		$13
Regulator	N/A			
CPU	Intel	NUC 6i7KYK	i7 3.5Ghz	$825
Internal Comm Network	TP-Link	5 Port Gigabit PoE Switch		$50
External Comm Innterface	NETGEAR	Nighthawk R7000		$143.75
Programming Language 1	Python			
Programming Language 2	C++			
Compass	built into Pixhawk and DVL			
Inertial Measurment Unit (IMU)	built into Pixhawk			
Doppler Velocity Log (DVL)	Teledyne	Workhorse Navigator 1200		
Camera(s)	Cameras: Edmund Optics: Allied Vision Mako G-234C Lenses: RMA Electronics: Tamron M112FM08			3 x $725 3 x $279
Hydrophones	Teledyne	TC4013		~$1000
Manipulator	N/A	custom 8021 aluminum, 3D-printing		
Algorithms: vision	ML Pipeline: Single Convolutional Neural Network Gate Detector: Segmentation, Thresholding, Hough Line Transform			
Algorithms: acoustics	Chebyshev filter, moving average, cross-correlation, Time Difference of Arrival			
Algorithms: localization and mapping	Extended Kalman Filter, SLAM w/ Stereo Cameras			
Algorithms: autonomy	N/A			
Open source software	Docker and ROS			
Team size (number of people)	32			
HW/SW expertise ratio	60/40 (19 HW, 13 SW)			
Testing time: simulation	10 hours			
Testing time: in-water	40			

Appendix C: Community Outreach

Mentoring: The Duke Robotics Club recognizes the importance of working with the local community and inspiring future generations of STEM students. The team has worked with both local middle schoolers and high schoolers, helping Durham Academy Middle School coach a pilot FIRST Lego League team and robotics afterschool program and mentoring Team 900 Zebracorns navigate the FIRST competition. The team has also advised countless teachers curriculums through a partnership with Project Lead the Way.

In 2018-2019, members of the Duke Robotics Club mentored a brand new team comprised of local Durham high school students, guiding them in creating a robot for the FIRST Tech Challenge, and also helping to bring them to the finals of the regional competition for the FIRST Robotics Competition. The members joined this FRC Team 6426 Robo Gladiators multiple times per week to help them strategize, build, and program their robot. Advancing to the finals of the regional competition as a rookie team, they outperformed many veteran teams. Looking forward to next season, the Robo Gladiators hope to extend their successes and become national champions, and the Duke Robotics Club looks forward to once again helping them succeed.

Even though each members time could have been spent improving Cthulhu, the team acknowledges how much more robotics can advance with each class of students. The Duke Robotics Club wants to encourage as much innovation as possible, and the team is proud to be able to spark ideas in generations of students to come.

Fig. 7. The local high schoolers' FIRST Robotics Competition team, mentored by two Duke Robotics Club freshmen, after reaching the regional finals in their pilot year.

Within Duke: The Duke Robotics Club has also spread robotics and STEM through outreach both within Duke's Pratt School of Engineering and the university as a whole. The team has run a Bot Battle competition to allow students of all disciplines to give robotics a try. The team hosts information sessions every semester, open to the entire university, attracting over 100 students each time. Lastly, the team has reached out to interest- and identity-based student groups on campus within STEM fields to coordinate cross-promotion and facilitate activities between members of related clubs.